Computational Learning Theory

Prof. Chia-Yu Lin

Yuan Ze University

2021 Spring

Thanks to the slides of Prof. Yu, Tian-Li from NTU.

Outline

- Sample Complexity
- Errors of a Hypothesis
- PAC Learnability
- Exhausting the Version Space
- Mistake Bounds

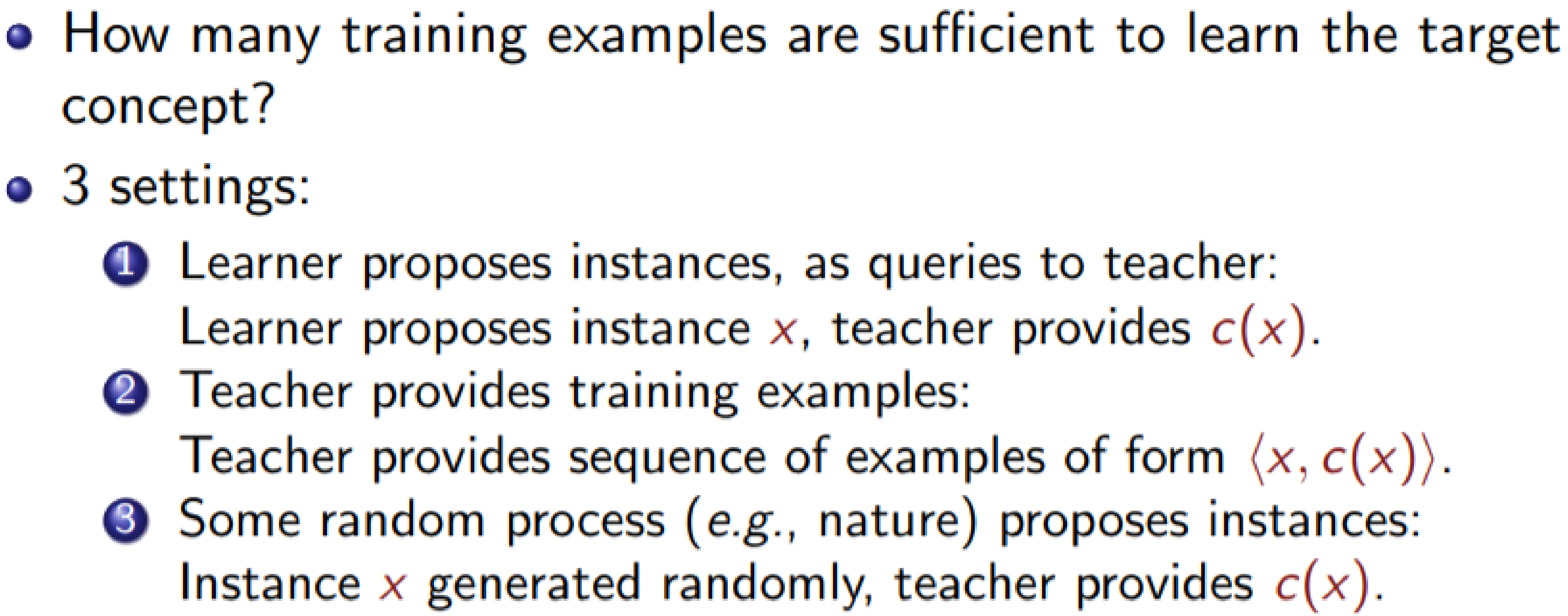
Computational Learning Theory

- What general laws constrain inductive learning?
- We seek theory to relate:
  - Complexity of hypothesis space considered by the learner
  - Accuracy to which target concept is approximated
  - Probability that the learner outputs a successful hypothesis
  - Manner in which training examples presented to the learner
- Goals:
  - Sample complexity: How many training examples are needed for successful learning?
  - Computational complexity: How much computational effort is needed for a learner to converge to a successful hypothesis?
  - Mistake bound: How many examples will the learner misclassify before the convergence?

Q1:

- Which of the following statements below is not the goal that computational learning theory want to achieve?
- (A) Learning successfully in polynomial time.
- (B) Finding out the upper and lower bound of error.
- (C) Deriving sample complexity.
- (D) All of the above.

Sample Complexity



- How many training examples are sufficient to learn the target concept?
- 3 settings:
  1. Learner proposes instances, as queries to teacher:
     Learner proposes instance x, teacher provides $c(x)$.
  2. Teacher provides training examples:
     Teacher provides sequence of examples of form $\langle x, c(x) \rangle$.
  3. Some random process (*e.g.*, nature) proposes instances:
     Instance x generated randomly, teacher provides $c(x)$.

Cross-validation

Sample Complexity: Setting 1

- Learner proposes instance x, teacher provides $c(x)$ (assume c is in learner's hypothesis space H)
- Optimal query strategy: play 20 questions
  - Pick instance x such that half of hypotheses in VS classify x positive, half classify x negative.
  - When this is possible, need $\underline{\lceil \log_2 |H| \rceil}$ queries to learn c. => Best case
  - When not possible, need even more.

Sample Complexity: Setting 2

- Teacher (who knows c) provides training examples (assume c is in learner's hypothesis space H)
- Optimal teaching strategy: depends on H used by learner.
- Consider the case where H is conjunctions of up to n boolean literals (positive or negative).
  - e.g., $(AirTemp = Warm) \wedge (Wind = Strong)$, where $AirTemp, Wind, \ldots$ each has 2 possible values.
  - if n possible boolean attributes in H, $(n+1)$ examples suffice.
  - Why?

The size of hypothesis space (|H|) : 3^n (Attribute is +, -, or ?)
The number of examples: log(|H|) => Worst case

如果concept有don 't care? (1/2)

	A_1	A_2	A_3	.....	A_n
Concept:	+	-	?	?...	?

要學會這樣的concept，需要提供幾個example??

Step1: 學don’ t care

A_1	A_2	A_3	.....	A_n	Class
+	-	+	+…	+	=> +
+	-	-	-…	-	=> +

需要兩個example來學所有的don't care

同時包含+ & - ，在conjunction做不到
=> 所以就會是don 't care

Step2: 學A_1只能是+ & A_2只能是-

+	+	+	+…	+	=> -
-	-	+	+…	+	=> -

如果concept有don 't care? (2/2)

A_1	A_2	A_3	…..	A_n
+	-	?	?…	?

要學會這樣的concept，需要提供幾個example??

Step1: 學don' t care

A_1	A_2	A_3	…..	A_n	Class
+	-	+	+…	+	=> +
+	-	-	-…	-	=> +

n-k

花兩個example來學k個don 't care

Step2: 學A_1只能是+ & A_2只能是-

+	+	+	+…	+	=> -
-	-	+	+…	+	=> -

n-k個 example

Total example: n-k+2. If there is don't care, k>=1 => n-k+2 <=n+1

如果concept都沒有don ‘t care?

	A_1	A_2	A_3	.....	A_n
Concept:	+	+	+	+...	+

要學會這樣的concept，需要提供幾個example??

A_1	A_2	A_3	.....	A_n	Class
+	+	+	+…	+	=> +
-	+	+	+…	+	=> -
+	-	+	+…	+	=> -
+	+	-	+…	+	=> -
⋮					

1 example

n example

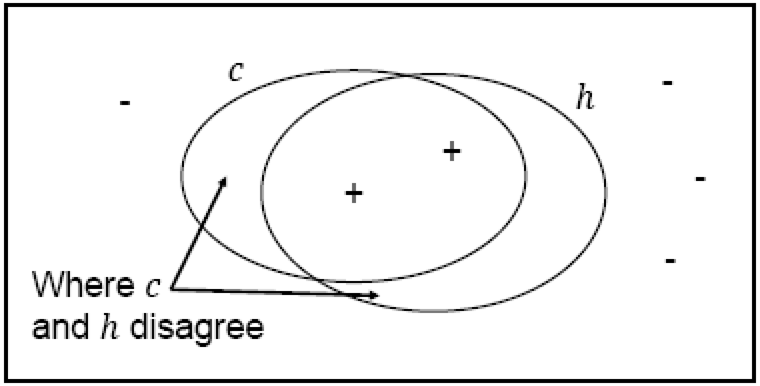
Total example: n+1

Sample Complexity: Setting 3

- **Given:**
  - Set of instances X.
  - Set of hypotheses H.
  - Set of possible target concepts C.
  - Training instances generated by a fixed, unknown probability distribution $\mathbb{D}$ over X.
- Learner observes a sequence D of training examples of form $\langle x, c(x) \rangle$, for some target concept $c \in C$.
  - Instances x are drawn from distribution $\mathbb{D}$.
  - Teacher provides target value $c(x)$ for each x.
- Learner must output a hypothesis h estimating c
  - h is evaluated by its performance on subsequent instances drawn according to $\mathbb{D}$
- **Note:** randomly drawn instances, noise-free classifications.

True Error of a Hypothesis

Instance Space X



Definition

The **true error** (denoted $error_{\mathbb{D}}(h)$) of hypothesis h with respect to target concept c and distribution $\mathbb{D}$ is the probability that h misclassifies an instance drawn at random according to $\mathbb{D}$.

$$error_{\mathbb{D}}(h) \equiv \Pr_{x \in \mathbb{D}} (c(x) \neq h(x))$$

Two Notations of Error

多常錯? =>100個training example 錯2個 =>2%

- Training error, denoted $error_{\underline{D}}(h)$, of hypothesis h with respect to c: How often $h(x) \neq c(x)$ over training instances.

機率

- True error, denoted $error_{\underline{\mathbb{D}}}(h)$, of hypothesis h with respect to c: How often $h(x) \neq c(x)$ over future random instances.

- Our concerns:

  Training error: 2% => True error不高於3%的機率是多少?

  - Can we bound the true error of h given its training error?
  - First consider when training error of h is zero ($i.e.$, $h \in VS_{H,D}$)

PAC Learning

- Consider a class C of possible target concepts defined over a set of instances X of length n, and a learner L using hypothesis space H.
- We desire that the learner probably learns a hypothesis that is approximately correct.

Definition

C is **PAC-learnable** by L using H if for all $c \in C$, distributions $\mathbb{D}$ over X, ϵ such that $\underline{0 < \epsilon < 1/2}$, and δ such that $\underline{0 < \delta < 1/2}$, learner L will with probability at least $\underline{(1 - \delta)}$ output a hypothesis $h \in H$ such that $\underline{error_{\mathbb{D}}(h) \leq \epsilon}$, in time that is polynomial in $1/\epsilon$, $1/\delta$, n and $size(c)$.

- To prove any concept is PAC-learnable or not, we need to derive the sample complexity needed for setting 3.

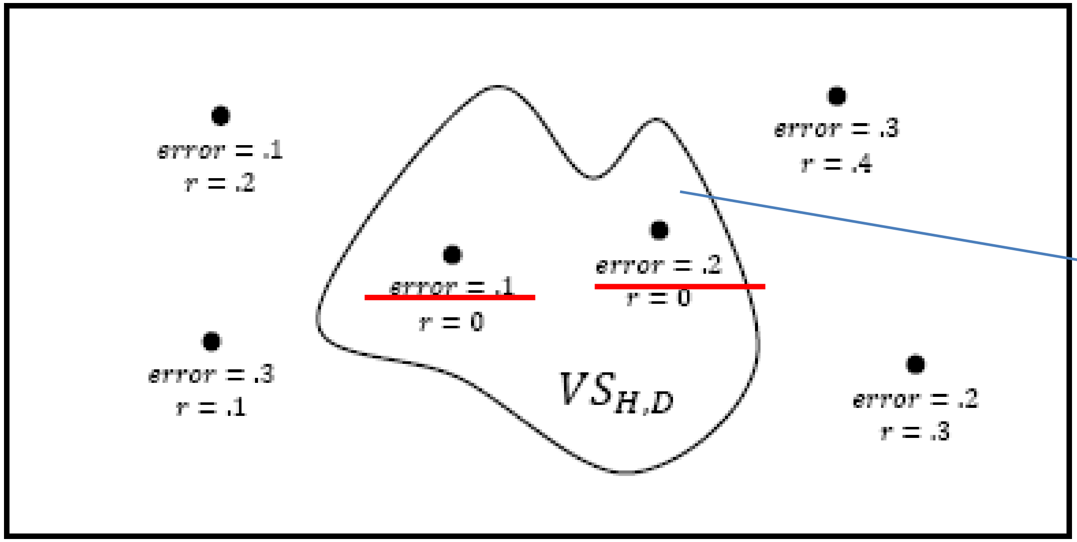
如果一個concept是PAC-learnable，代表此concept沒有很難，可以在夠短的時間內， 夠高的機率輸出一個夠準確的hypothesis

Q2:

- Which of the following statements is true about PAC learning?
- (A) The parameters ε should be less than ½.
- (B) The algorithm is expected to output a hypothesis that is approximately correct.
- (C) If the concept is PAC learnable, we can get an accurate hypothesis with a high enough probability in a short time.
- (D) All of the above.

Exhausting the Version Space

Hypothesis Space H



**

r: training error
error: true error

This version space is **0.3-exhausted**.

(r is training error, $error$ is true error)

Definition

The version space $VS_{H,D}$ is ϵ-**exhausted** with respect to c and $\mathbb{D}$, if every hypothesis h in $VS_{H,D}$ has error less than ϵ with respect to c and $\mathbb{D}$.

$$(\forall h \in VS_{H,D})\ error_{\mathbb{D}}(h) < \epsilon$$

所有

Question

- Given training error is 0 (i.e. hypothesis is in version space), what is the true error?

- => How many examples can make version space

ε-exhausted?

Probability of Exhausting the Version Space

- How many examples ϵ-exhaust the VS?

Theorem (Haussler, 1988)

If H is finite, and D is a sequence of $m \geq 1$ independent random examples (from distribution $\mathbb{D}$) of some target concept c, then for any $0 \leq \epsilon \leq 1$, the probability that $VS_{H,D}$ is not ϵ-exhausted is less than or equal to

$$|H|e^{-\epsilon m}.$$

- The above theorem bounds the probability that any consistent learner will output a hypothesis h with $error_{\mathbb{D}}(h) \geq \epsilon$.
- If we want to this probability to be below δ

$$|H|e^{-\epsilon m} \boxed{\leq \delta} \overset{\log}{\Rightarrow} m \geq \frac{1}{\epsilon}(\ln|H| + \ln(1/\delta))$$

1-δ的機率輸出夠準確的
hypothesis
所需要的example

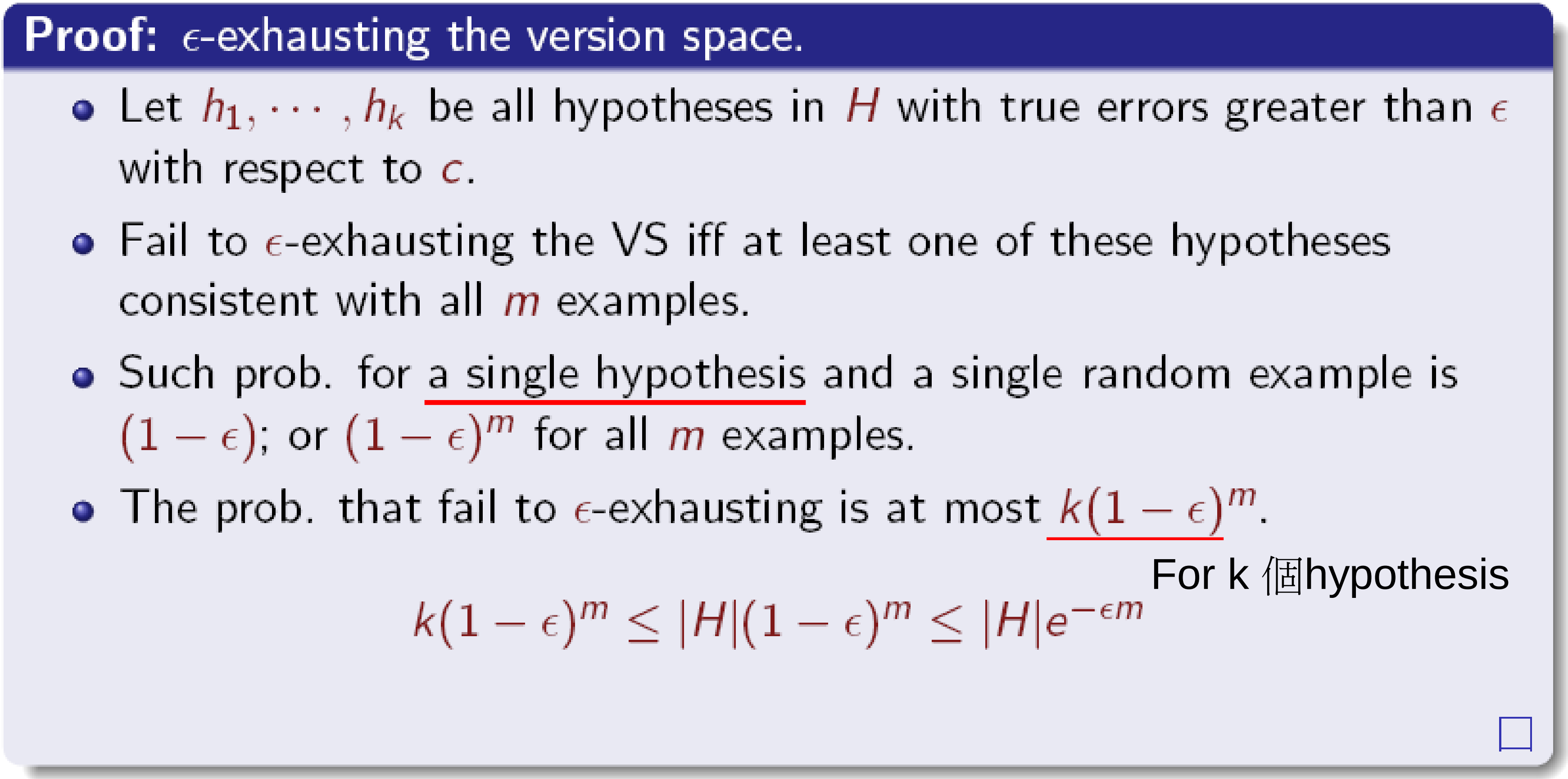
充分但不必要條件!!

Q3:

- Which of the following statements is true about the probability of the version space is not ε-exhausted?
- (A) By this theorem , we can know the most number of example drawn from distribution, that we can get a hypothesis such that the true error is large than or equal to ε.
- (B) According to this, we can infer that if, Pr will be large than or equal to $|H|e^{-\varepsilon m}$.
- (C) m is the symbol of the number of the examples.
- (D) The theorem is still true, if H is infinite.

Proof of ε-exhausting (1/2)

- What is the probability that version space is not ε-exhausted if m examples are given?



Proof: ϵ-exhausting the version space.

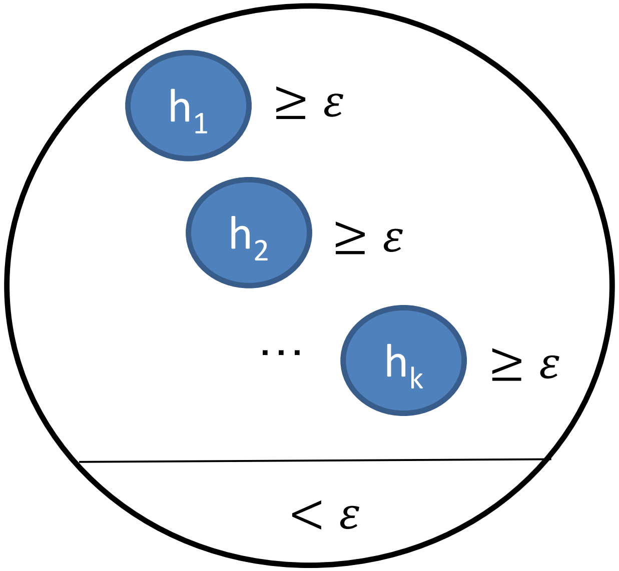
- Let $h_1, \cdots, h_k$ be all hypotheses in H with true errors greater than ϵ with respect to c.
- Fail to ϵ-exhausting the VS iff at least one of these hypotheses consistent with all m examples.
- Such prob. for a single hypothesis and a single random example is $(1-\epsilon)$; or $(1-\epsilon)^m$ for all m examples.
- The prob. that fail to ϵ-exhausting is at most $\underline{k(1-\epsilon)}^m$.

For k 個hypothesis

$$k(1-\epsilon)^m \leq |H|(1-\epsilon)^m \leq |H|e^{-\epsilon m}$$

□

Proof of ε-exhausting (2/2)



k個h $error_{ID}(h_i) \geq \varepsilon$

$h(x_1)$: +
$c(x_1)$: +

h has to consistent with c
Otherwise, h is not in the version space.
The probability of h consistent with c based on x_1 is $1-\varepsilon$

$h(x_2)$: -
$c(x_2)$: -

The probability of h consistent with c based on x_2 is $1-\varepsilon$

⋮ m examples

After asking m times, the probability of h consistent with c is $(1-\varepsilon)^m$

Learning Conjunctions of Boolean Literals

- Recall that $m \geq \frac{1}{\epsilon}(\ln|H| + \ln(1/\delta))$ examples are sufficient to assure with probability at least $(1-\delta)$ that every h in $VS_{H,D}$ satisfies $error_{\mathbb{D}}(h) \leq \epsilon$.

- Suppose H contains conjunctions of constraints on up to n boolean attributes.
  - $|H| = 3^n$. Every attribute can be (+, -, don't care)
  - $m \geq \underline{\frac{1}{\epsilon}}(\underline{n}\ln 3 + \ln\underline{(1/\delta)})$
  - Boolean conjunctions is PAC-learnable!

Polynomial in $\frac{1}{\varepsilon}$.

Polynomial in $\frac{1}{\delta}$.

Polynomial in n

EnjoySport Revisit

- Inn *EnjoySport*, if we consider only conjunctions, $|H| = 973$.

$$m \geq \frac{1}{\epsilon}(\ln 973 + \ln(1/\delta))$$

- If want to assure that with probability 95%, *VS* contains only hypotheses with $error_{\mathbb{D}}(h) \leq 0.1$, then it is sufficient to have m examples, where

$$m \geq \frac{1}{0.1}\left(\ln 973 + \ln \frac{1}{0.05}\right)$$

$$m \geq 98.8$$

⇒ m=99 就充分

⇒ 給99個example，就有95%以上的機率可以輸出一個true error<10%的hypothesis

Agnostic Learning (Learning Inconsistent Hypotheses)

- The equation $m \geq \frac{1}{\epsilon}(\ln|H| + \ln(1/\delta))$ tells us how many training examples suffice to ensure that every hypotheses in H having zero training error will have true error of at most ϵ.

C ≠H

- However, if $\boxed{c \notin H}$, zero training error may not be achievable.
- We desire to know how many examples suffice to ensure $error_{\mathbb{D}}(h) \leq error_D(h) + \epsilon$.
- **Hoeffding bounds:** $|\bar{X} - \mu|$

$$\Pr\left(error_{\mathbb{D}}(h) > error_D(h) + \epsilon\right) \leq e^{-2m\epsilon^2}$$

- Sample complexity in this case:

$$\Pr\left((\exists h \in H)\ error_{\mathbb{D}}(h) > error_D(h) + \epsilon\right) \leq \underline{|H|e}^{-2m\epsilon^2} \leq \delta$$

$$m \geq \underline{\frac{1}{2\epsilon^2}}(\ln|H| + \ln(1/\delta))$$

H個

Infinite Hypothesis Space

- The above sample complexity has two drawbacks:
  1. Weak bounds.
  2. H has to be finite.
- We need another measure of the complexity of H.

Definition

A **dichotomy** of a set S is a partition of S into two disjoint subsets.

Definition

A set of instances S is **shattered** by hypothesis space H iff for every dichotomy of S there exists some hypothesis in H consistent with this dichotomy.

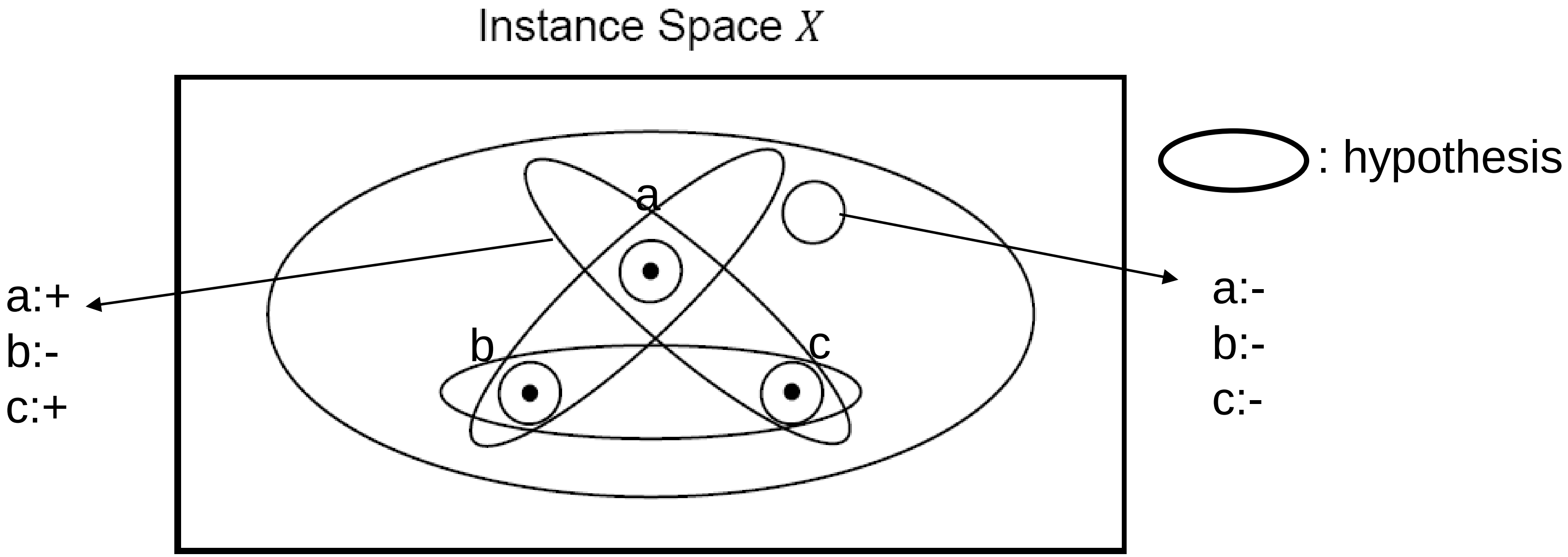
$$S=\{a,b,c\} \Rightarrow \left.\begin{array}{l}\{a\}\\ \{b,c\}\end{array}\right\} h \in H \quad \{a\}:+ \quad \{b,c\}:-$$

Shattering a Set of Instances (1/2)

- S is a subset of instances, $S \subseteq X$; $2^{|S|}$ distinct dichotomies in total.
- Each $h \in H$ imposes a dichotomy on S:

$$\{x \in S | h(x) = 0\} \text{ and } \{x \in S | h(x) = 1\}$$

- H shatters S iff every dichotomy of S is represented by some $h \in H$.



a, b, c instances have 8 dichotomies.

=>如果8個dichotomies對應的h都在H裡
=>S is shattered by H

Shattering a Set of Instances (2/2)

- H shatter S => $|H| \geq 2^{|S|}$

a	b	C	
+	+	+	h_1
+	+	-	h_2
…			…
-	-	-	h_8

8個h
均屬於H

The Vapnik-Chervonenkis (VC) Dimension

- The ability to shatter a set of instances is closely related to the inductive bias of the hypothesis space.
- An unbiased hypothesis space can represent every possible concept (dichotomy) over X: An unbiased hypothesis space shatters X.
- What if H cannot shatter X, but can shatter a subset S?
- Intuitively, the larger S is, the more expressive H is.

Definition

The **Vapnik-Chervonenkis dimension**, $VC(H)$, of hypothesis space H is the size of the largest finite subset of instance space X shattered by H. If arbitrarily large finite sets of X can be shattered by H, then $VC(H) \equiv \infty$.

- Note that for any finite H, $VC(H) \leq \log_2 |H|$.

=>$|H| \geq 2^{|S|}$ => $|H| \geq 2^{|VC(H)|}$
=>雙邊取log

Why VC Dimension?

- Make VC dimension to define sample complexity.
- Since $m \geq log$|H| is too weak, we will use VC Dimension to bound.

Q4:

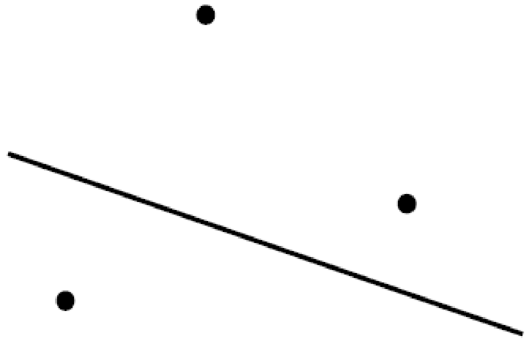
- Which of the following statements is the application of VC dimension?
- (A) The complexity of the model.
- (B) The accuracy of the prediction.
- (C) The speed of the computation.
- (D) The upper bound of the training examples.

VC Dimension (1/3)

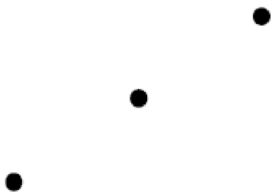
- Instances are real numbers: $X = \mathbb{R}$
- Hypotheses are real intervals: $h_{ab} = a < x < b$; $H = \{\forall a, b\ h_{ab}\}$
- Consider $S = \{3.1, 5.7\}$. H shatters S, why?
- For any set of 3 instances: $S = \{x, y, z\}$, where $x < y < z$. There is no way for H to represent this dichotomy: $\{x, z\}$ and $\{y\}$.

$$VC(H) = 2$$

- For 2D points (X) and line separations (H), $VC(H) = 3$.



(a)

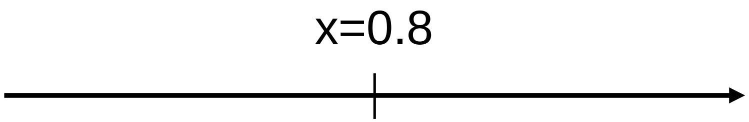


(b)

Example: 1 Instance on a Line

$X = \mathbb{R}$

$/H/ = \infty$



{x} => Dichotomy: $\emptyset, \{x\}$

$\{x\}, \emptyset$

Is there h can make $\emptyset: +, \{x\}: -$? =>don' t include x: $h_{10,20}$

Is there h can make$\{x\}: +, \emptyset: -$? =>include x: $h_{0,1}$

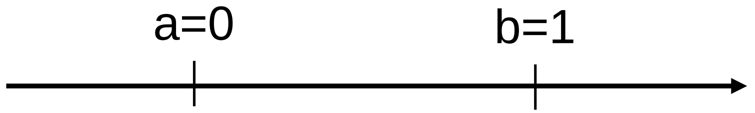
$h_{10,20}$ and $h_{0,1}$ are belong to H => H shatter {x}

VC(H)=? $VC(H) \geq 1$

Example: 2 Instances on a Line

$X = \mathbb{R}$

$|H| = \infty$



Dichotomy: 4 => + +
+ -
- +
- -

Is there h can $\text{get} \ + \ +$? => Include a and b: $h_{5,5}$

Is there h can $\text{get} \ + \ -$? =>Include a and not include b: $h_{-5,0.5}$

Is there h can $\text{get} \ - \ +$? =>not include a and include b: $h_{0.5,5}$

Is there h can $\text{get} \ - \ -$? =>not include a and b: $h_{20,40}$

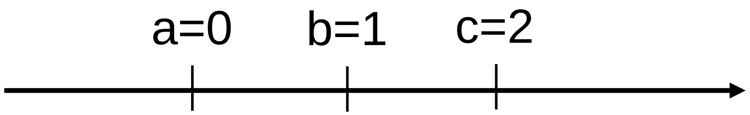
All h are belong to H => H shatter {a,b}

VC(H)=? $VC(H) \geq 2$

Example: 3 Instances on a Line

$X = \mathbb{R}$

$|H| = \infty$



Dichotomy: 8

Is there h can $\mathrm{get}\ + \ - \ +$? => Include a, c and not include b:??

=> We cannot get a “h” to shatter **any** 3 instances in the line.

By definition of VC, we have to shatter “every” dichotomy

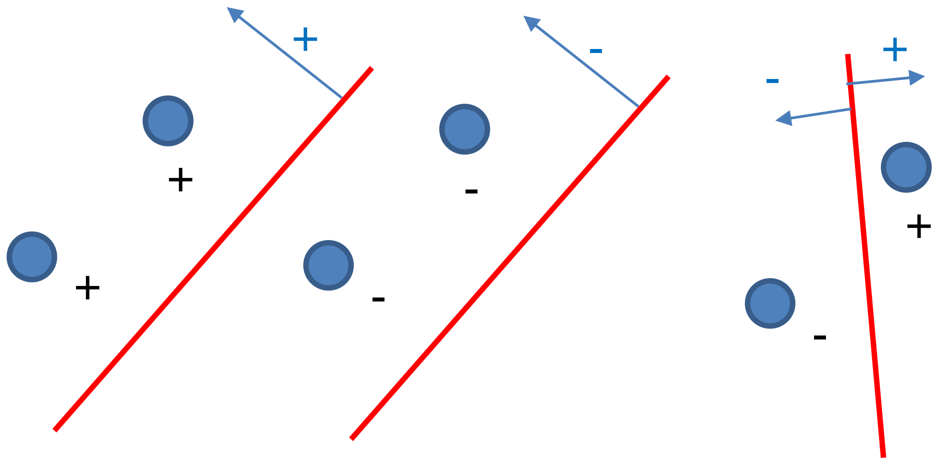
=> $\mathrm{VC(H)} \neq 3$

=> $VC(H) = 2$

Example: Linear Classifier with 2 Instances

$X = \mathbb{R}^2 = \{(x,y) | x,y \in R\}$

$m(H) = \{(x,y) | ax+by+c \geq 0,\ a,b,c \in R\}$



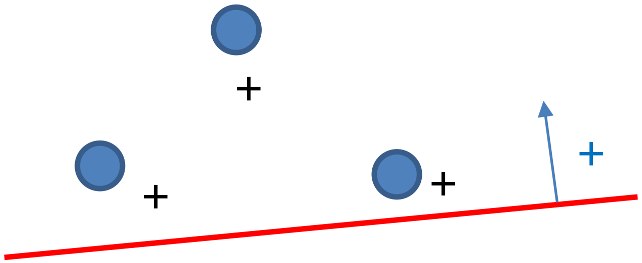
VC(H)=?

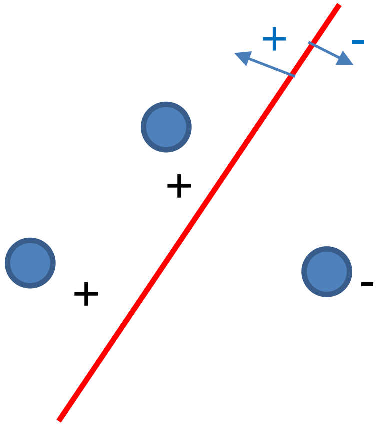
$\Rightarrow VC(H) \geq 2$

Example: Linear Classifier with 3 Instances

$X = \mathbb{R}^2 = \{(x,y) | x,y \in R\}$

$m(H) = \{(x,y) | ax+by+c \geq 0, a,b,c \in R\}$





VC(H)=?

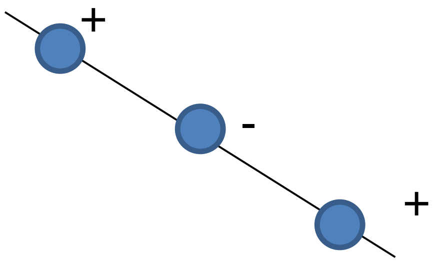
$\Rightarrow VC(H) \geq 3$

Example: Linear Classifier with 3 Instances

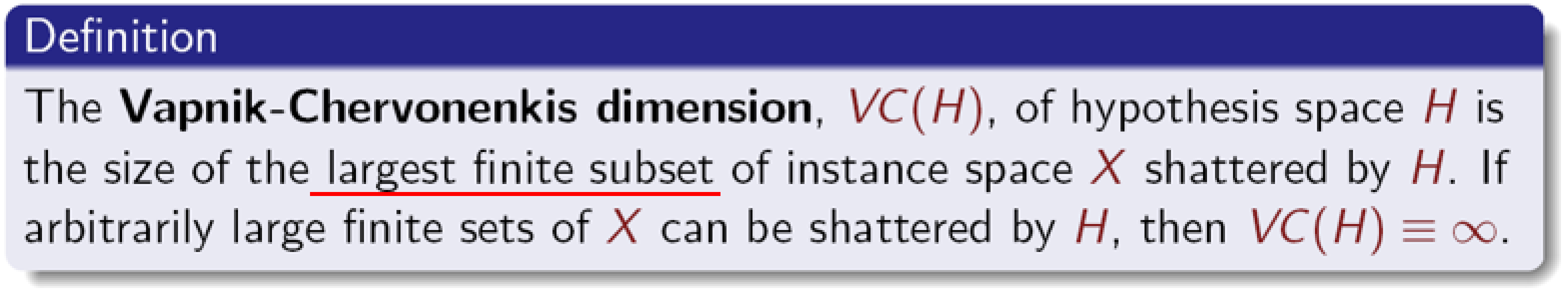
$$X = \mathbb{R}^2 = \{(x,y) | x,y \in R\}$$

$$m(H) = \{(x,y) | ax+by+c \geq 0,\ a,b,c \in R\}$$

If 3 instances are on a line??



We cannot find a linear classifier to shatter 3 instances on a line.
So $VC(H) \geq 2$??



Definition

The **Vapnik-Chervonenkis dimension**, $VC(H)$, of hypothesis space H is the size of the largest finite subset of instance space X shattered by H. If arbitrarily large finite sets of X can be shattered by H, then $VC(H) \equiv \infty$.

Q5:

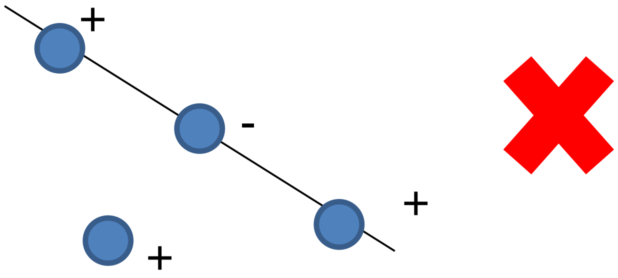
- Consider the case on the 2D plane. VC(H)=?
- (A) 2
- (B) 3
- (C) 4
- (D) 8

Example: Linear Classifier with 4 Instances

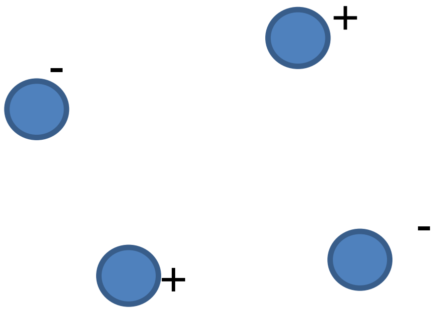
$X = \mathbb{R}^2 = \{(x,y) | x,y \in R\}$

$m(H) = \{(x,y) | ax+by+c \geq 0, a,b,c \in R\}$

Case 1: Any 3 instances are on a line.



Case 2: Any 3 instances are not on a line.



Dichotomy: 16

⇒ There is one dichotomy cannot be shattered.

⇒ XOR problem.

VC(H)=?

$=> \mathrm{VC(H)} \neq 4$

$=> VC(H) = 3$

Linear Classifier in n Dimension

- Linear classifier in n dimension => In general, the VC is n+1